In the Specification

Please amend the specification of this application as follows:

Rewrite the paragraph at page 4, lines 26 to 29 as follows:

--2. Four sets of control bits are provided. Two sets are for the source address including ~~Source Word Size~~ source word size and source increment size. Two sets are for the target address including target word size and target increment size.--

B1

Rewrite the paragraph at page 7, lines 16 to 29 as follows:

--ARM has also proposed use of a second bus for isolation and using a single arbiter. As shown in Figure 1, this second bus is called the advanced peripheral bus (APB) 120. APB bus 120 operates in the same fashion as AHB bus 100. APB bus 120 is connected to AHB bus 100 via an AHB-to-APB bus bridge 109. AHB-to-APB bus bridge 109 is a slave to AHB bus 100. The two bus system with single M-bus arbiter 110 is of limited usefulness, because it allows only one transaction to progress at a given time period. Note that all high performance devices including memory and high performance peripheral device 130 are on AHB bus 100. All peripheral devices of moderate performance including UART 115, timer 116, keypad 117 as well as peripherals 121 to ~~123~~ 122 reside on the peripheral bus 120.--

B2

Rewrite the paragraph at page 7, line 30 to page 8, line 29 as follows:

--Figure 2 illustrates the signal flow between a master requesting control of the AHB bus, the arbiter performing the arbitration decision and the slave selected by the master for a command to be executed in this standard AMBA system. AHB ~~bust~~ bus arbiter 111, AHB master 200 and AHB slave 210 each receive a reset signal ~~HResetx~~ HReset 222 and a clock signal ~~HClockx~~ HClock 223.

B3

The AHB master 200 makes the request of AHB arbiter 110 by activating HBusReqx signal 231. The AHB master 200 receives permission from AHB arbiter 110 by HGrantx signal 232. The AHB master 200 confirms the grant and locks this arbitration decision by ~~HLock~~ HLockx signal 233. AHB master 200 then sends address 205 to AHB decoder 111. AHB decoder 111 activates a select signal 112 supplied to the selected slave device. In this example the selected slave device is AHB slave 210. The interaction of AHB master 200 and AHB slave 210 is completed via the control signals 213 and acknowledged via HResp signal 211 and HReady signal 212. Data for read and write operations flows between all masters and all slaves via the AHB bus 100. AHB slave 210 supplies data to AHB bus 100 via HRData[31:0] bus 206 and receives data from AHB bus 100 via HWData[31:0] bus 207. Likewise, AHB master 200 receives data from AHB bus 100 via HRData[31:0] bus 208 and supplies data to AHB bus 100 via HWData[31:0] bus 209. Note in this regard that reads and writes are considered from the point of view of AHB master 200. Thus in a data read data flows from AHB slave 210 to AHB bus 100 via HRData[31:0] bus 206 and from AHB bus 100 via HRData[31:0] bus 208. Of course only one master is activated at a given time and this master selects only one slave ~~on~~ with which it will execute a transfer (read or write) command.

Rewrite the paragraph at page 13, line 24 to page 14, line 8 as follows:

Regarding DMA data byte shifter 304, consider the transfer of a larger word such as 32-bit word to a destination having a smaller word size such as 8-bits. In this type of operation ~~multiple,~~ multiple writes need to be made to the destination. On the first clock, the lower byte or half word needs to be aligned to the lower byte or half word in order to be transferred to the destination. On the next clock, byte 2 is written, byte 3 moves to byte 1

position etc. When all four bytes are written, micro-controller DMA can make the next read from the 32-bit source location and repeat the process. With 32-bit registers in both the read data register 301 and the write data register 310, the micro-controller DMA has sufficient flexibility through its control outputs 332 to slave devices to implement the multiple writes required for the full range of total and partial data transfers required.--

Rewrite the paragraph at page 15, lines 3 to 14 as follows:

--CNTVAL counter 333 is examined to determine it is hexadecimal zero (0x0000) (decision block 408). If DMA data transfer has not completed the entire DMA transfer yet (no branch 408), that is CNTVAL counter 333 has not yet reached 0x0000, then the DMA controller will execute the next word of the transfer. The initial source and target addresses (block 411) are incremented (processing block ~~411~~ 406). These addresses are initialized according to start source address register SADDR 325 and start target address register TADDR 326. The source address is incremented by the value stored in source address increment size register SIS 323 and the target address is incremented by the value stored in target increment size register TIS 324.--